

COMP 9517 Computer Vision

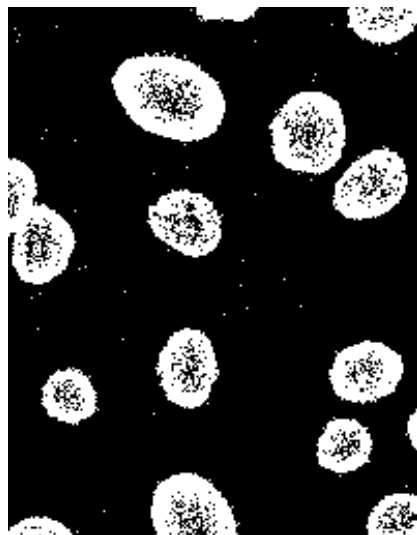
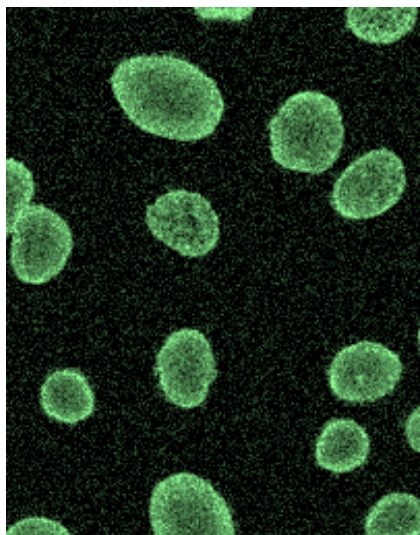
Segmentation

Part 2

# How to improve image segmentation results?

---

## Processing using mathematical morphology



How to clean up background noise?

How to clean up object noise?

How to separate touching objects?

How to close holes in objects?

How to extract object contours?

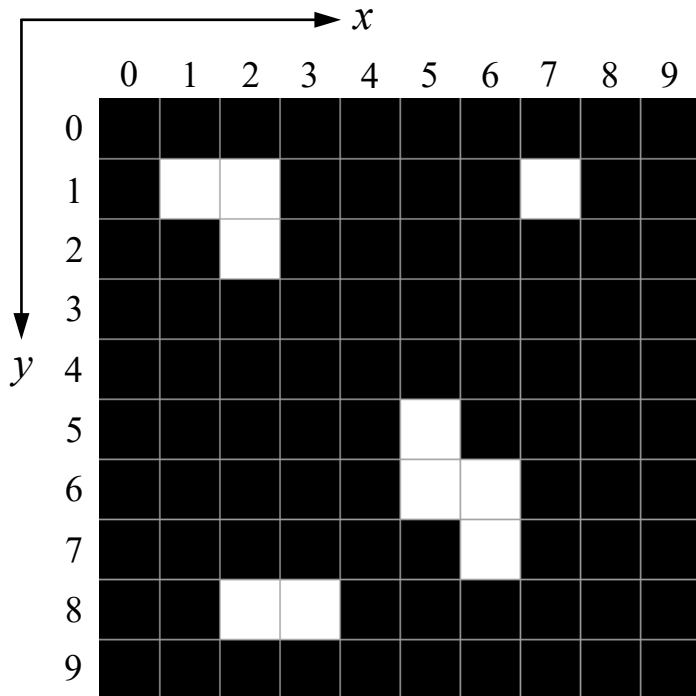
How to compute distances?

...

- Binary mathematical morphology (applies to binary images)
- Gray-scale mathematical morphology (applies to gray-scale images)

*Nonlinear image processing based on set-theory rather than on calculus*

# Binary image representations



**Binary image**

white = foreground  
black = background

0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

**Image as matrix**

1 = foreground  
0 = background

$$I = \{(1,1), (2,1), (7,1), (2,2), (5,5), (5,6), (6,6), (6,7), (2,8), (3,8)\}$$

**Image as set**

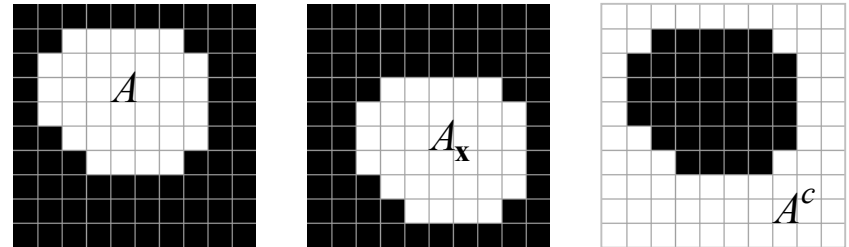
# Basic set operations

**Given sets**  $A = \{\mathbf{a}_i\}_{i=1}^{N_A}$  **and**  $B = \{\mathbf{b}_i\}_{i=1}^{N_B}$ , **with**  $\mathbf{a}_i, \mathbf{b}_i \in \mathbb{Z}^n$ , **we have**

Translation:  $A_{\mathbf{x}} = \{\mathbf{c}_i \mid \mathbf{c}_i = \mathbf{a}_i + \mathbf{x}, \mathbf{a}_i \in A\}$  for any given  $\mathbf{x} \in \mathbb{Z}^n$

Reflection:  $A^r = \{\mathbf{x}_i \mid \mathbf{x}_i = -\mathbf{a}_i, \mathbf{a}_i \in A\}$

Complement:  $A^c = \{\mathbf{x}_i \mid \mathbf{x}_i \notin A\}$

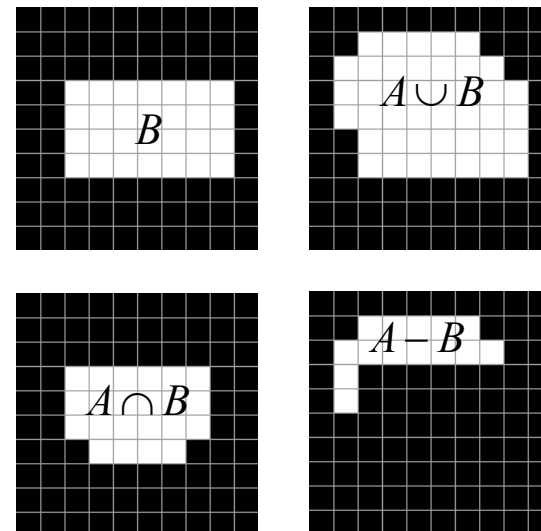


Union:  $A \cup B = \{\mathbf{x}_i \mid \mathbf{x}_i \in A \vee \mathbf{x}_i \in B\}$

Intersection:  $A \cap B = \{\mathbf{x}_i \mid \mathbf{x}_i \in A \wedge \mathbf{x}_i \in B\}$

Difference:  $A - B = \{\mathbf{x}_i \mid \mathbf{x}_i \in A \wedge \mathbf{x}_i \notin B\}$

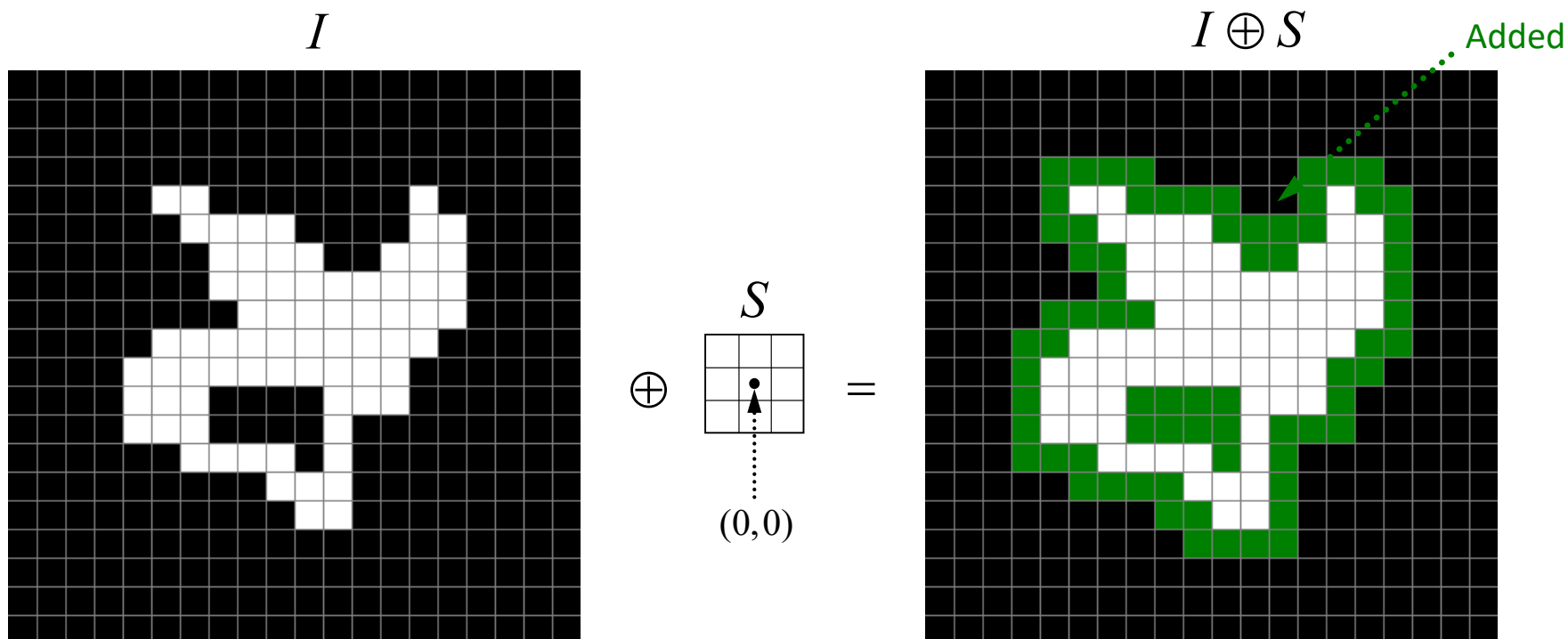
Cardinality:  $|A| = N_A$  and  $|B| = N_B$



# Dilation of binary images

**Definition of binary dilation:**  $I \oplus S = \left\{ \mathbf{x} \mid S_{\mathbf{x}}^r \cap I \neq \emptyset \right\}$

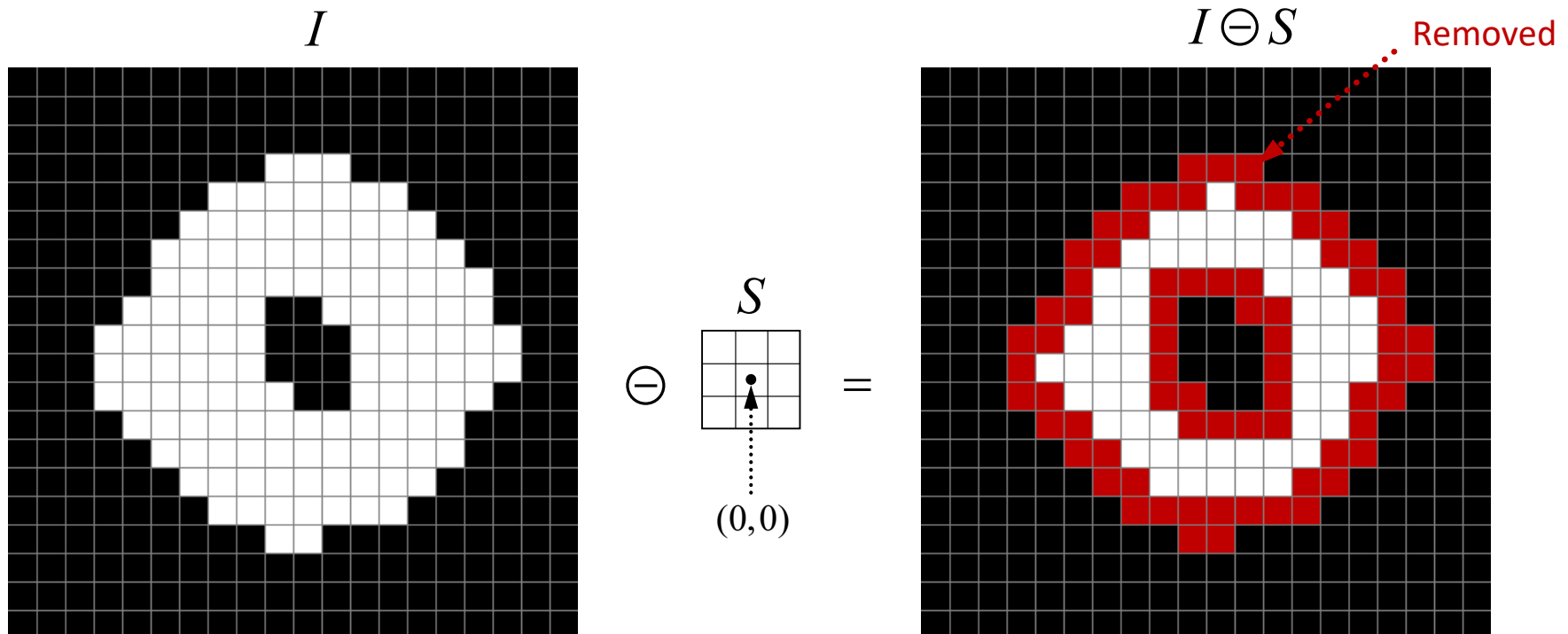
That is, the set of points  $\mathbf{x} \in \mathbb{Z}^n$  for which the intersection of the image  $I$  with the reflected version of a *structuring element*  $S$  translated to  $\mathbf{x}$ , is not empty



# Erosion of binary images

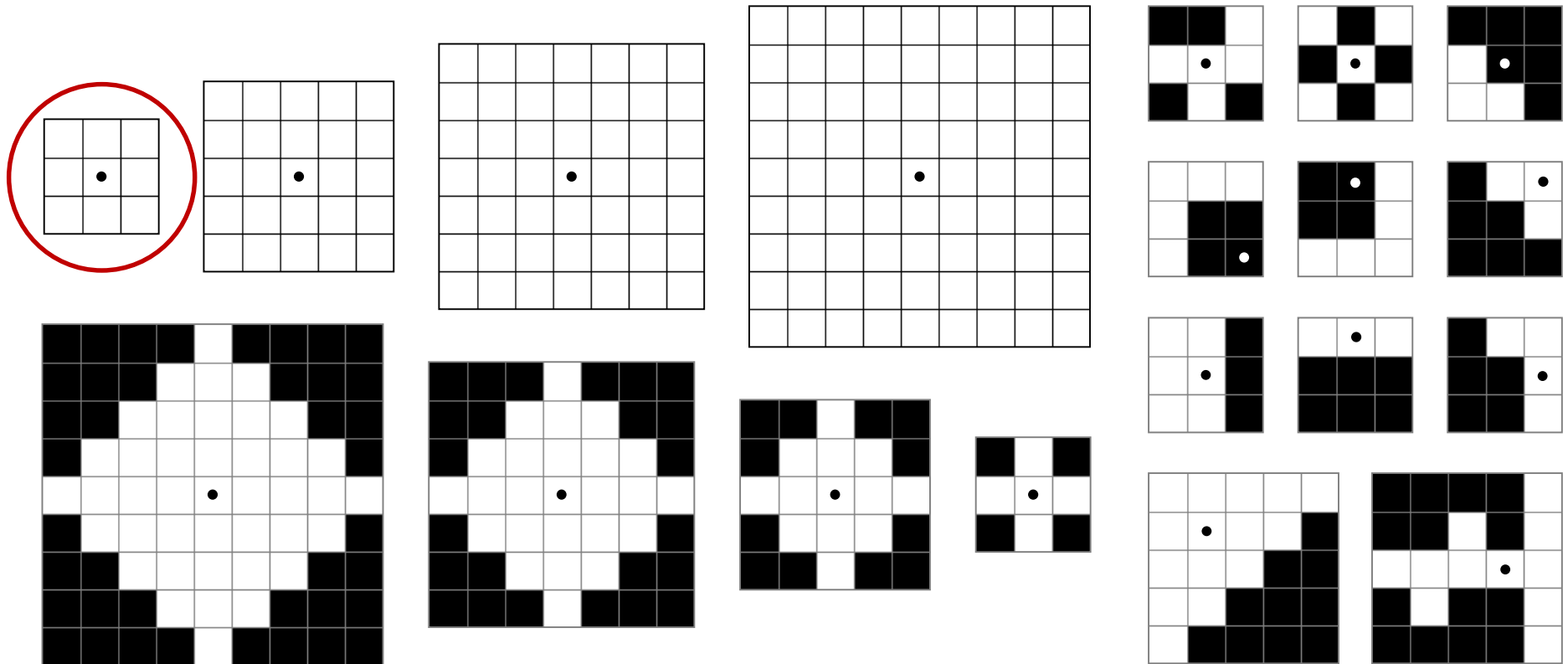
**Definition of binary erosion:**  $I \ominus S = \{\mathbf{x} \mid S_{\mathbf{x}} \subseteq I\}$

That is, the set of points  $\mathbf{x} \in \mathbb{Z}^n$  for which the *structuring element*  $S$  translated over  $\mathbf{x}$  is completely contained in the image  $I$



# Structuring elements

In principle a structuring element can have any shape ...



... but the symmetric 3 x 3 structuring element is used most often

# Dimensional decomposition

## Decomposition of the basic structuring element

$$\begin{aligned}
 I \oplus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} &= \left( I \oplus \begin{array}{|c|c|c|} \hline & \bullet & \\ \hline & & \\ \hline \end{array} \right) \oplus \begin{array}{|c|} \hline \\ \hline \bullet \\ \hline \\ \hline \end{array} & \quad I \ominus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} &= \left( I \ominus \begin{array}{|c|c|c|} \hline & \bullet & \\ \hline & & \\ \hline \end{array} \right) \ominus \begin{array}{|c|} \hline \\ \hline \bullet \\ \hline \\ \hline \end{array} \\
 \text{Dilation} & & \text{Erosion}
 \end{aligned}$$

$$\begin{aligned}
 I \oplus \begin{array}{|c|c|c|c|c|c|c|c|} \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & \bullet & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline & & & & & & & \\ \hline \end{array} &= I \oplus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} \oplus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} \oplus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} \oplus \begin{array}{|c|c|c|} \hline & & \\ \hline & \bullet & \\ \hline & & \\ \hline \end{array} \\
 & \text{Applies to dilation and erosion}
 \end{aligned}$$



# Opening of binary images

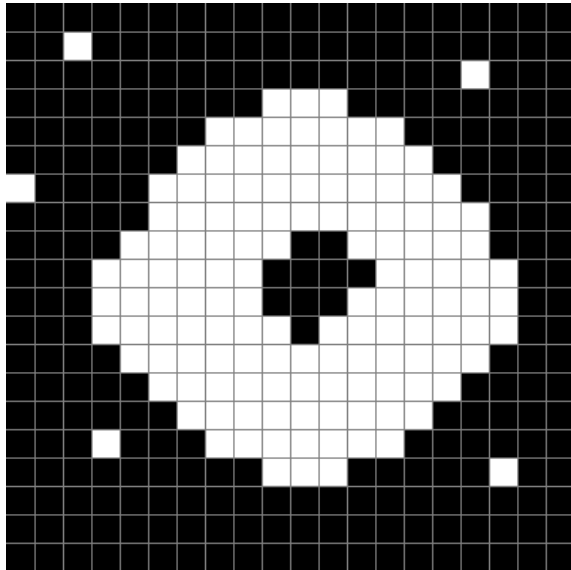
---

**Definition of binary opening:**  $I \circ S = (I \ominus S) \oplus S$

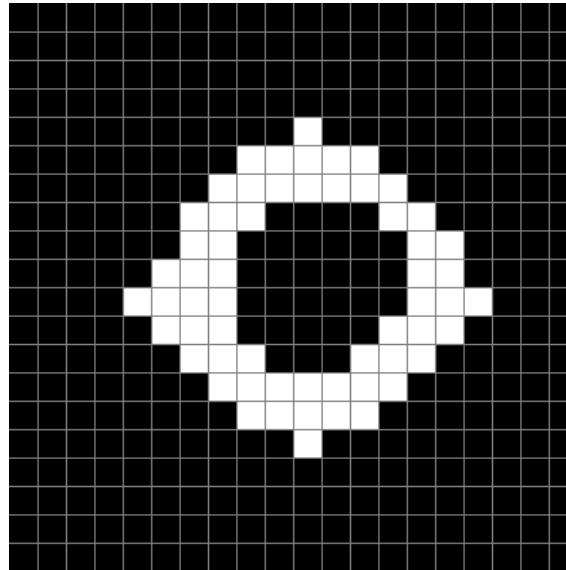
That is, an erosion followed by a dilation using the same structuring element

Example using the basic 3 x 3 structuring element:

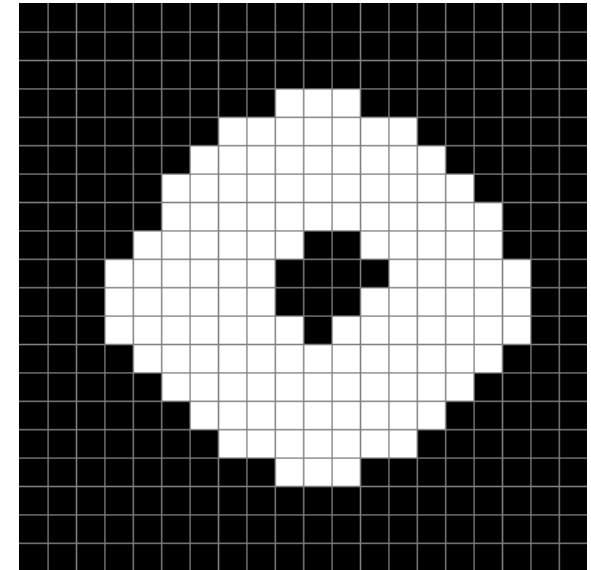
$I$



$I \ominus S$



$I \circ S$



Eliminates details smaller than the structuring element *outside* the main object

# Closing of binary images

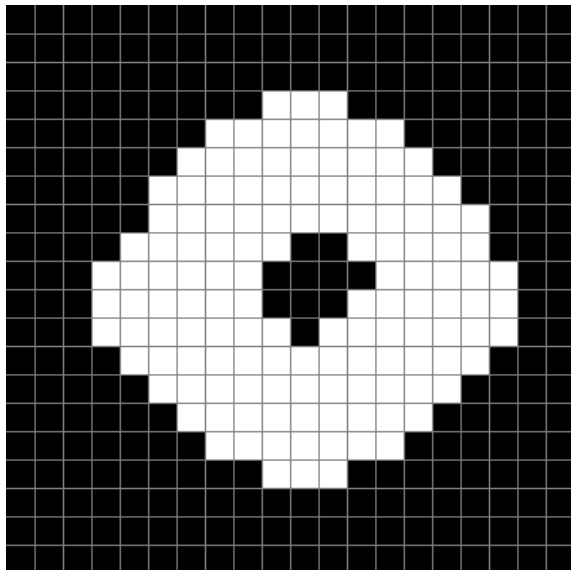
---

**Definition of binary closing:**  $I \bullet S = (I \oplus S) \ominus S$

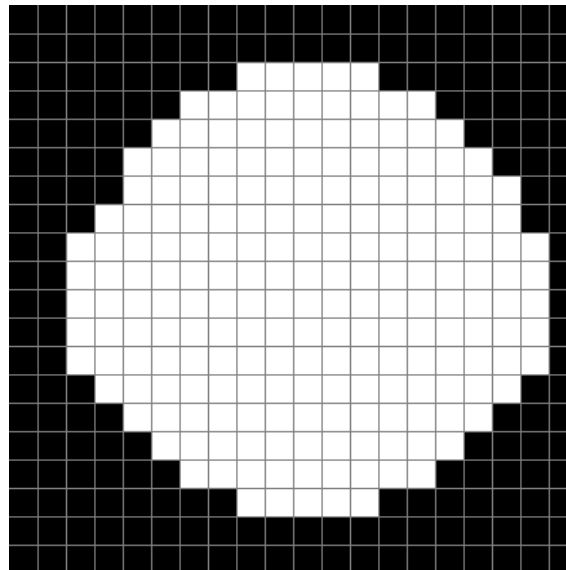
That is, a dilation followed by an erosion using the same structuring element

Example using the basic 3 x 3 structuring element:

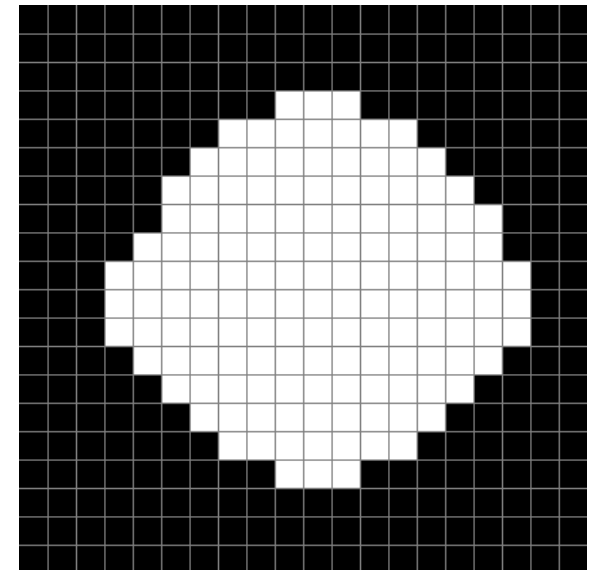
$I$



$I \oplus S$



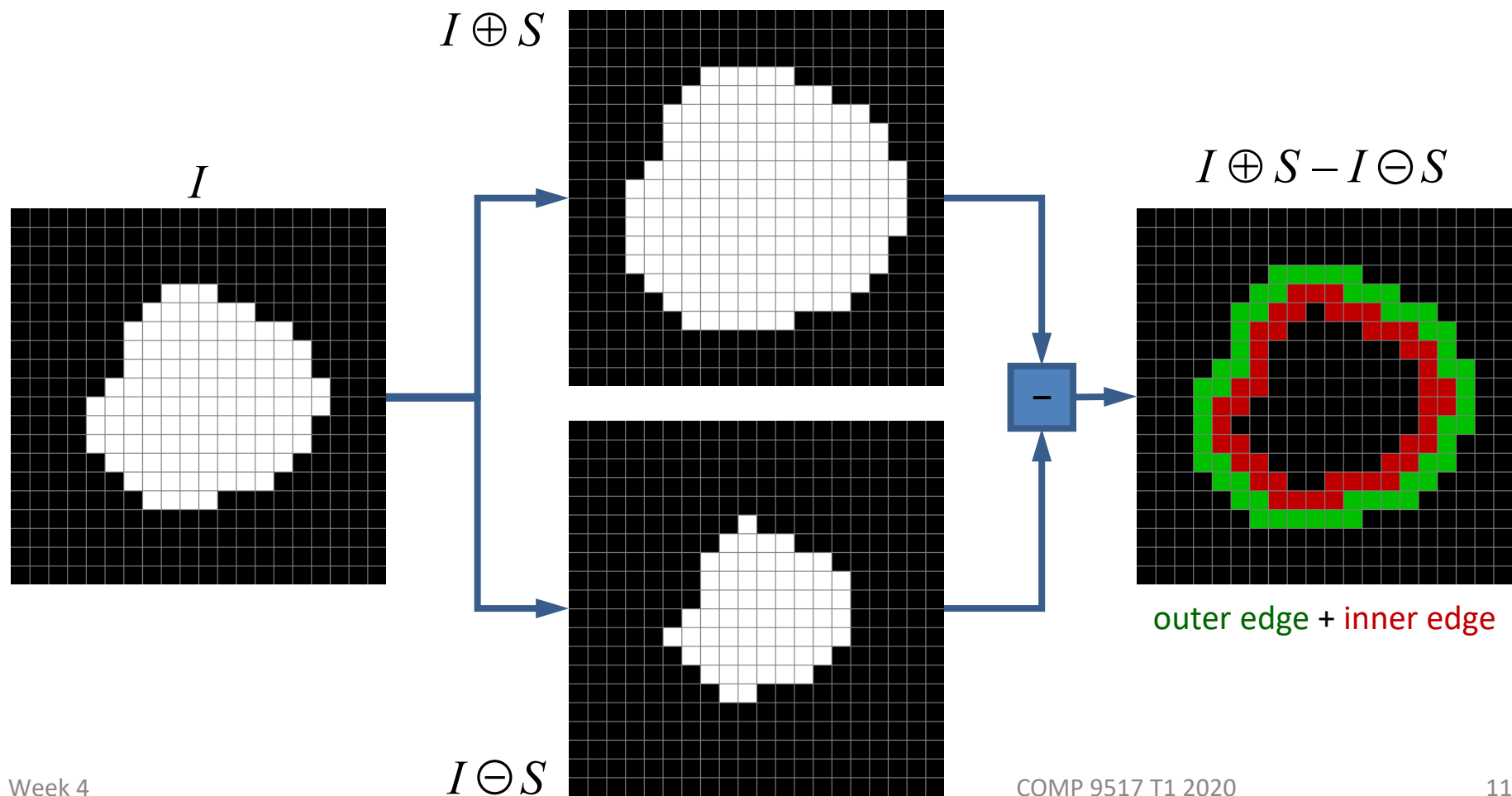
$I \bullet S$



Eliminates details smaller than the structuring element *inside* the main object

# Morphological edge detection

## Difference between the dilated and the eroded image

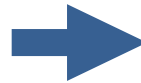
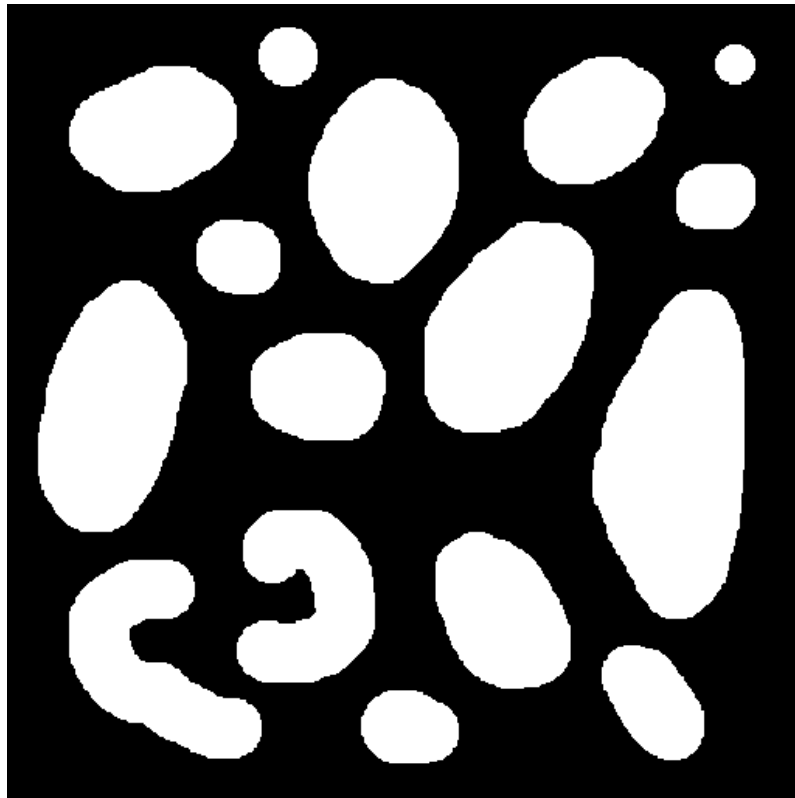


# Binary object outlines

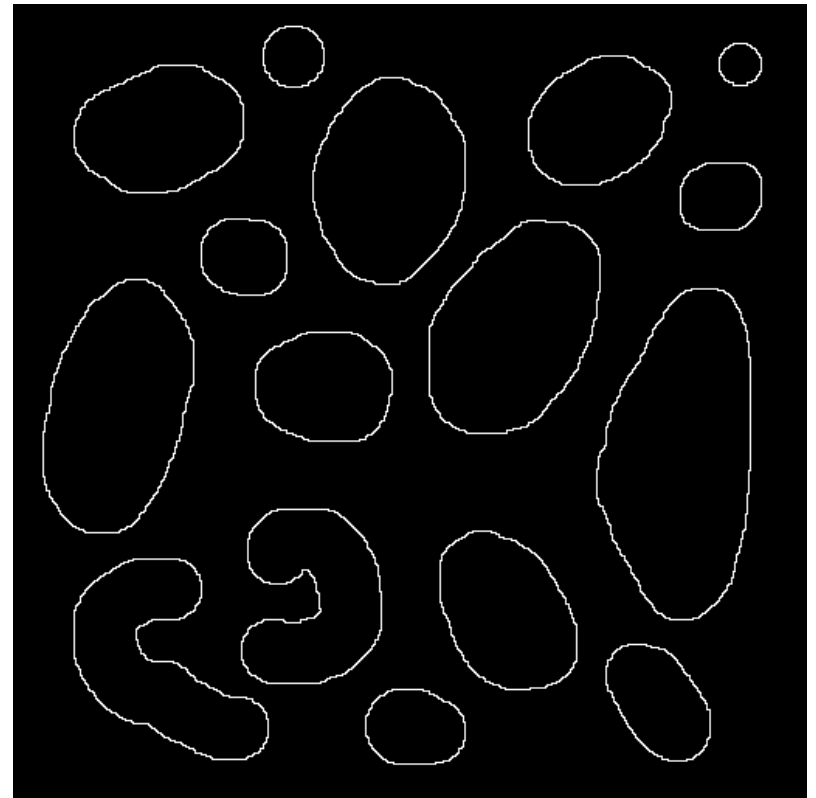
---

How to get a one-pixel thick outline of all objects in the image?

$I$



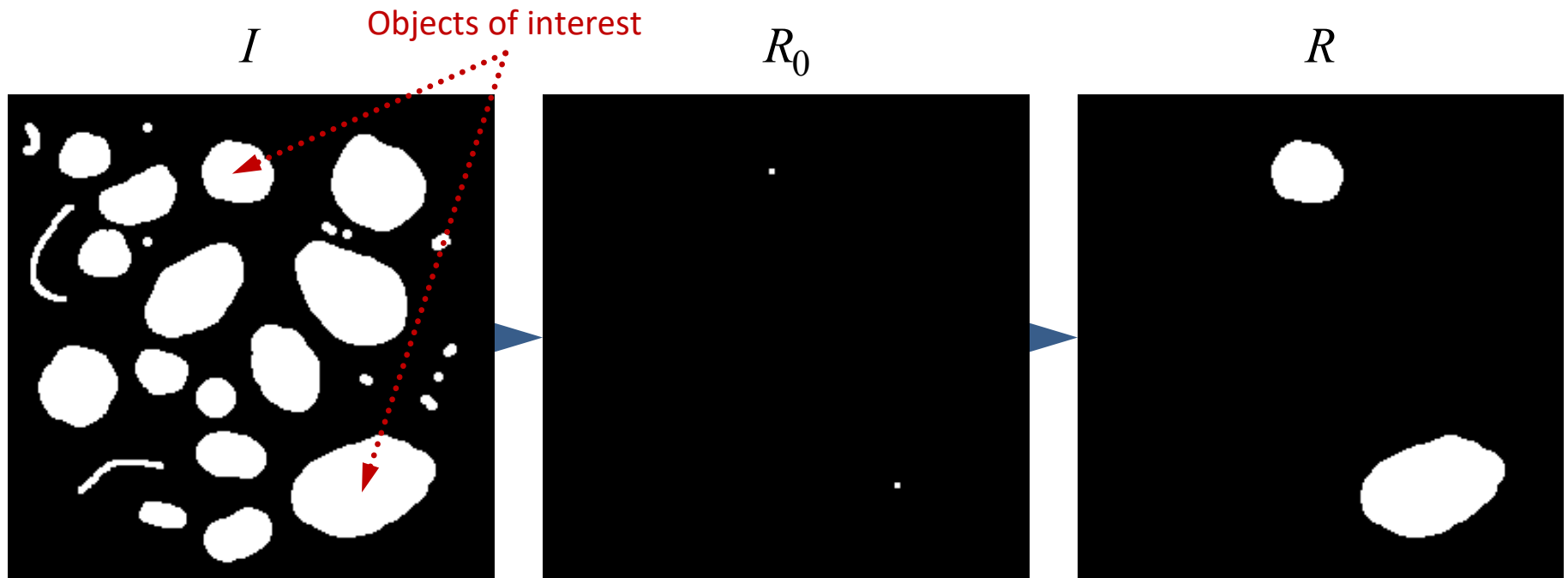
$I \oplus S - I$



# Reconstruction of binary objects

## How to create an image containing selected objects only?

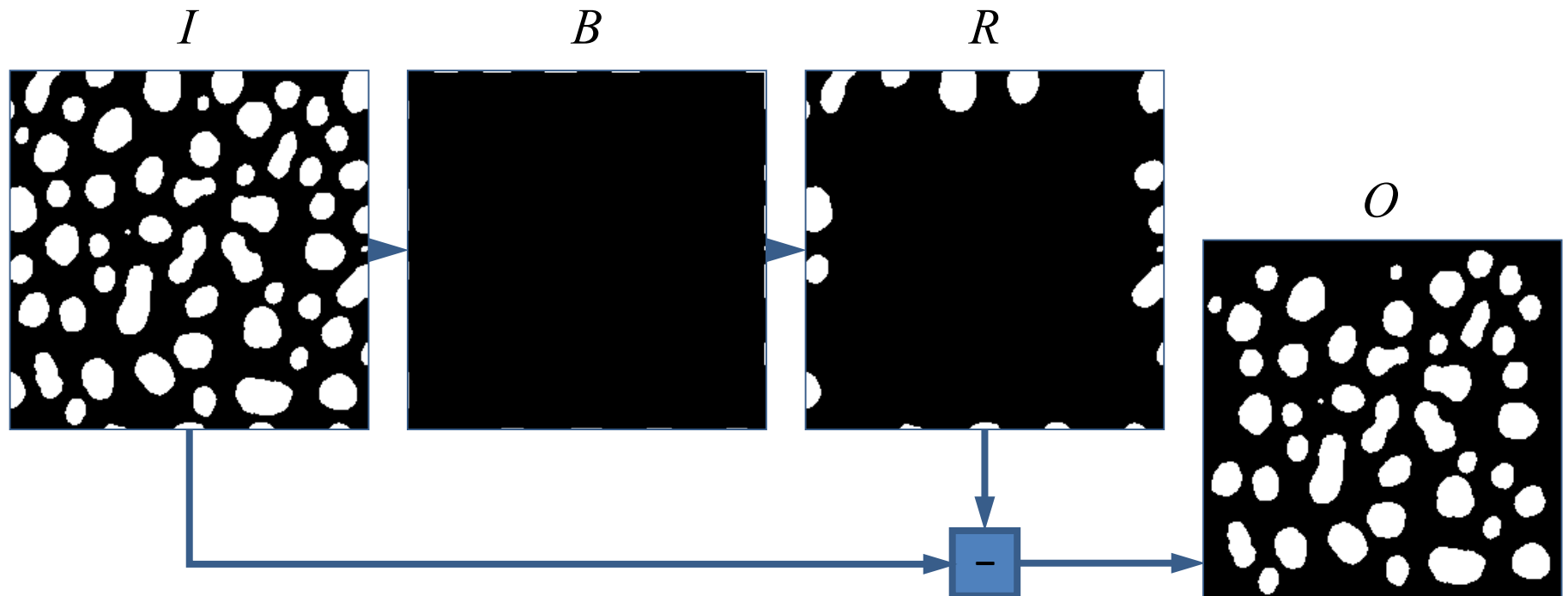
Create a marker image  $R_0$  containing seed pixels from each selected object of image  $I$  and then iteratively compute  $R_i = (R_{i-1} \oplus S) \cap I$  until  $R_i = R_{i-1}$



# Reconstruction of binary objects

## How to remove objects that are only partly in the image?

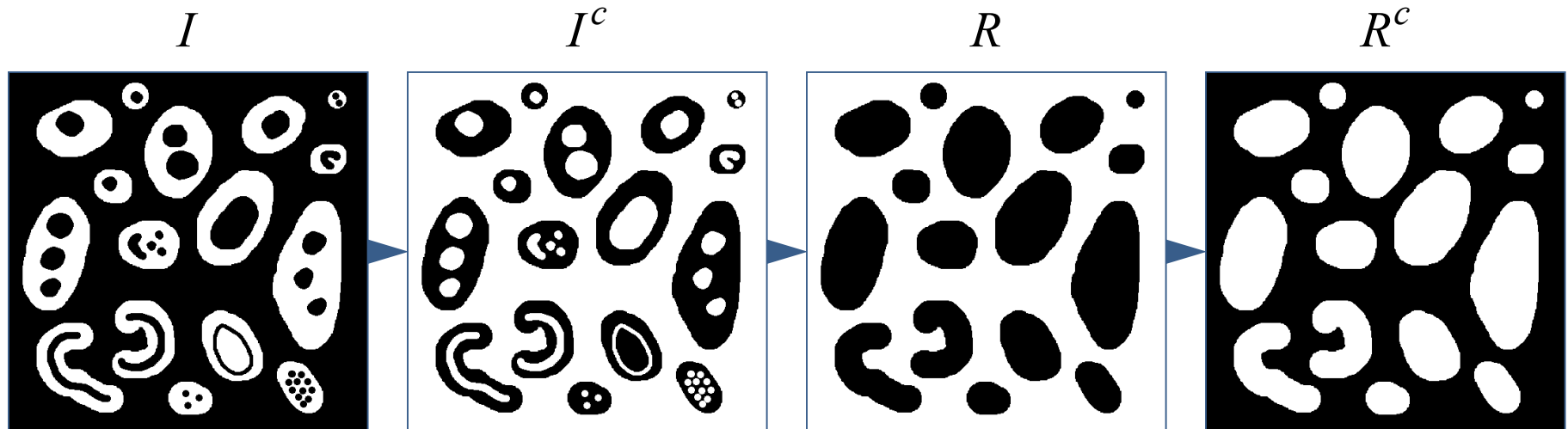
Take the boundary pixels  $B$  of input image  $I$  as the seeds, compute the reconstruction  $R$  from those seeds, and subtract the result from the input



# Reconstruction of binary objects

## How to fill all holes in all of the objects in the image?

Take the complement  $I^c$  of image  $I$ , take the boundary pixels of  $I^c$  as seeds, compute reconstruction  $R$  of  $I^c$  from those seeds, take the complement  $R^c$

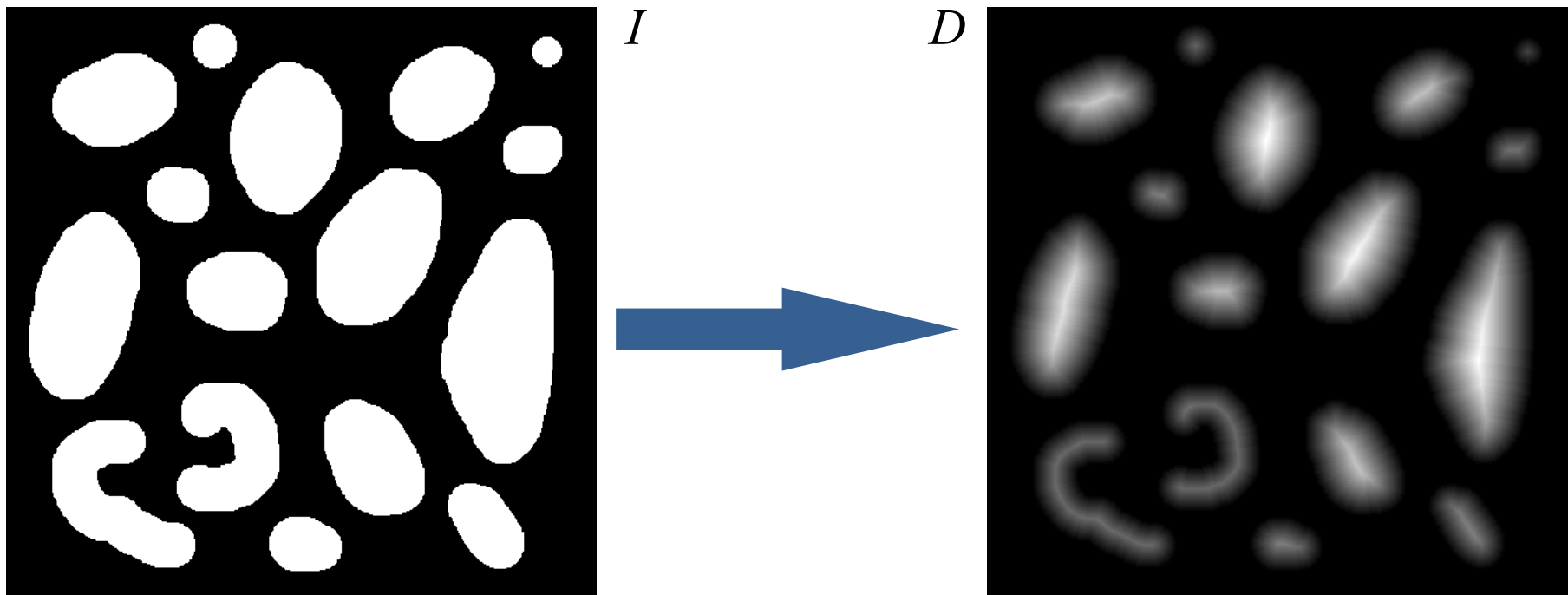


# Distance transform of binary images

---

## How to compute the distance of object pixels to the background?

Denote input image  $I$  as  $I_0$  and iteratively compute  $I_i = I_{i-1} \ominus S$  for  $i = 1, 2, \dots$  while setting all pixels eroded in iteration  $i$  to value  $i$  in the output image  $D$





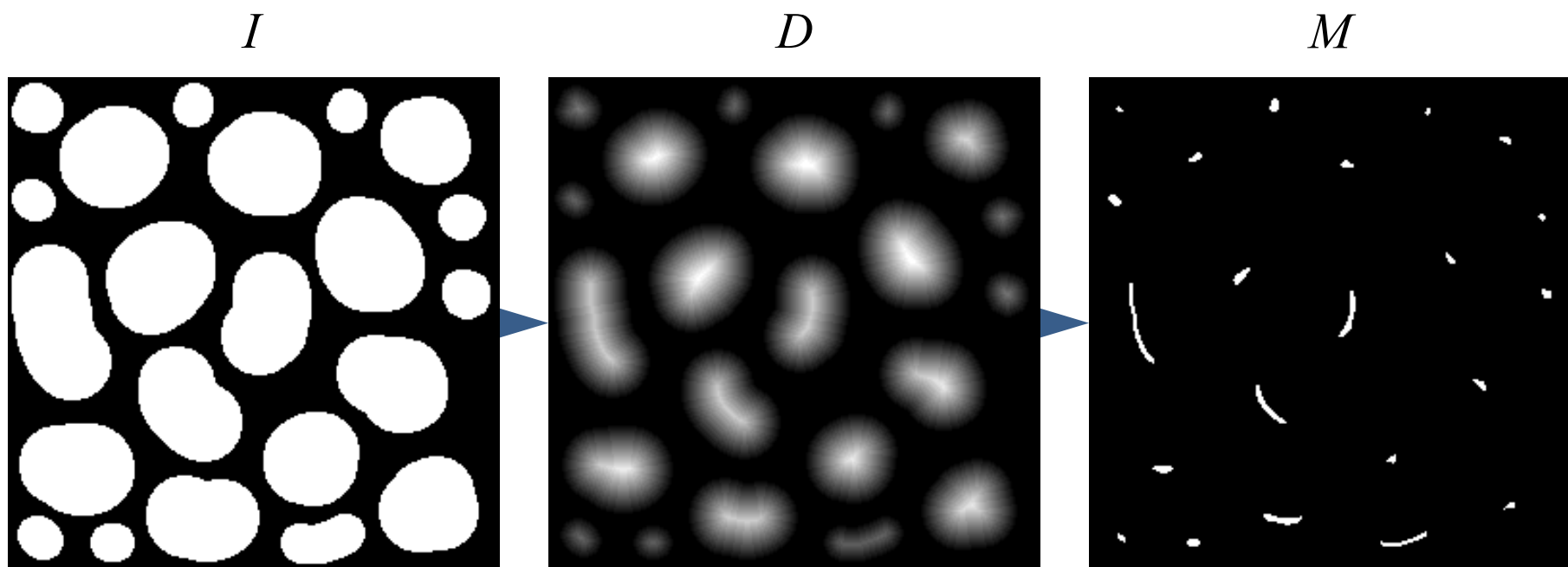
# Ultimate erosion of binary images

---

## How to find representative center points for all the objects?

Compute the distance transform of the image and find all the local maxima

This is the same as keeping only the last object pixels before final erosion

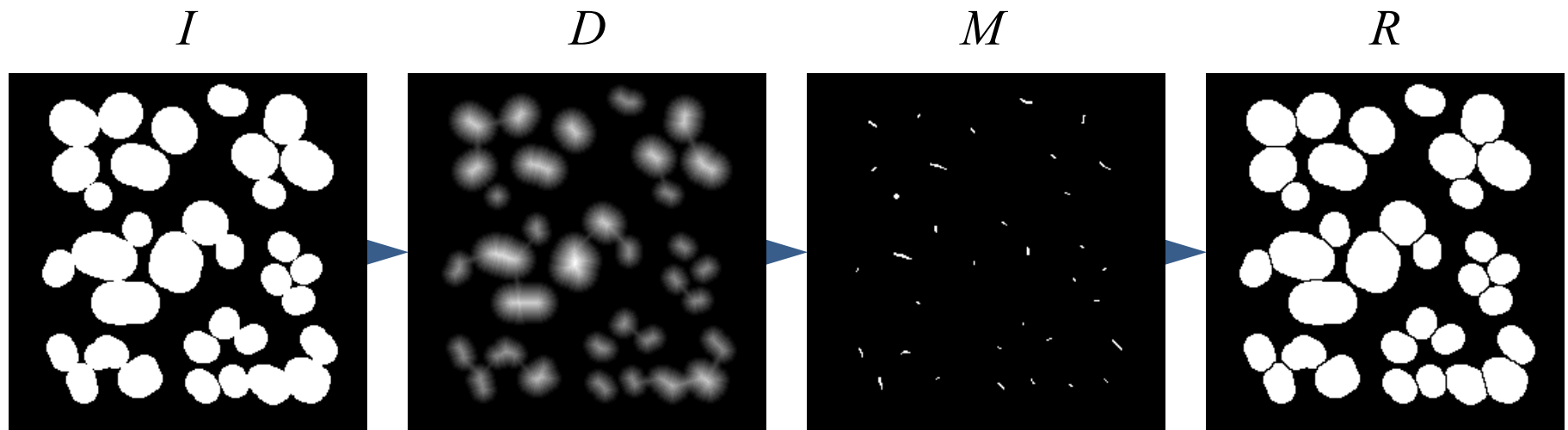


# Ultimate erosion and reconstruction

---

## How to separate objects that are touching each other?

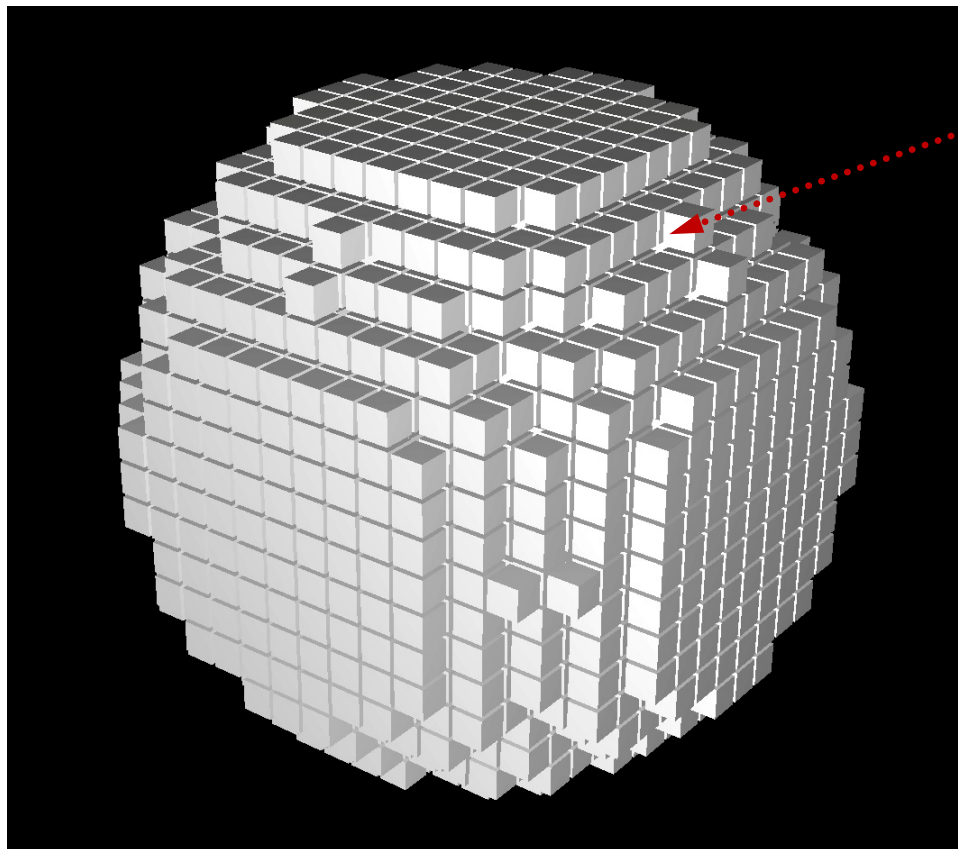
Perform ultimate erosion and then perform a reconstruction of the result with the additional constraint that objects may not merge



Since elongated objects may result in multiple local maxima this approach works best for objects that are more or less circular

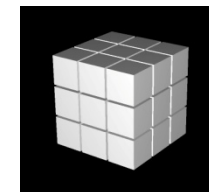
# Binary morphology of $n$ D images

The presented concepts extend to any dimensionality



Example of a 3D binary image

*Volumetric pixels ("voxels")*

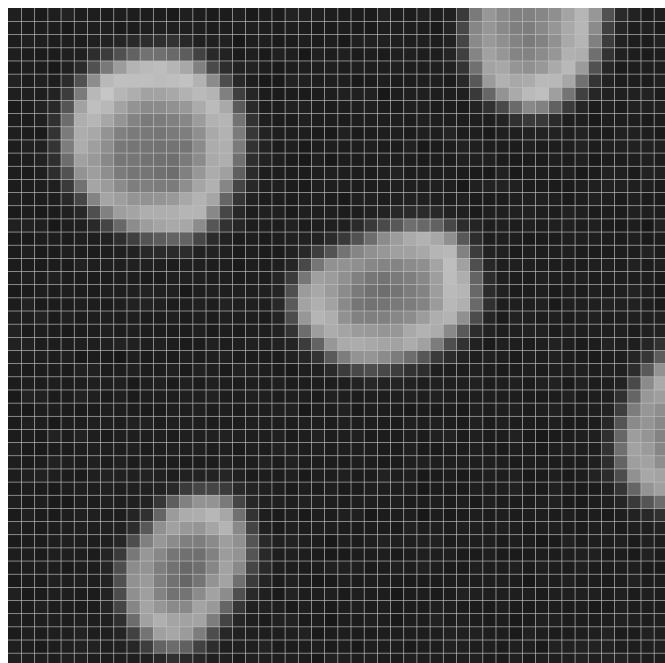


3 x 3 x 3 voxel  
structuring  
element

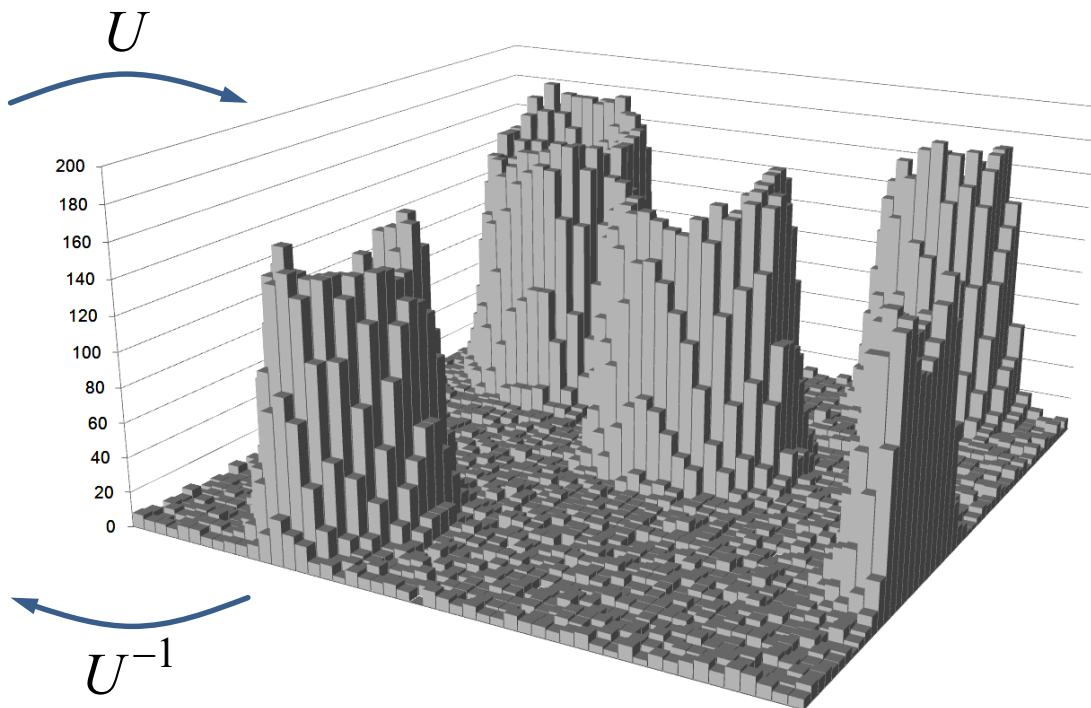
- 3D dilation
- 3D erosion
- 3D opening
- 3D closing
- And all algorithms based on it

# Gray-scale mathematical morphology

Consider  $n$ D gray-scale images as  $(n+1)$ D binary images...



2D gray-scale image  $I$



3D binary image  $U(I)$

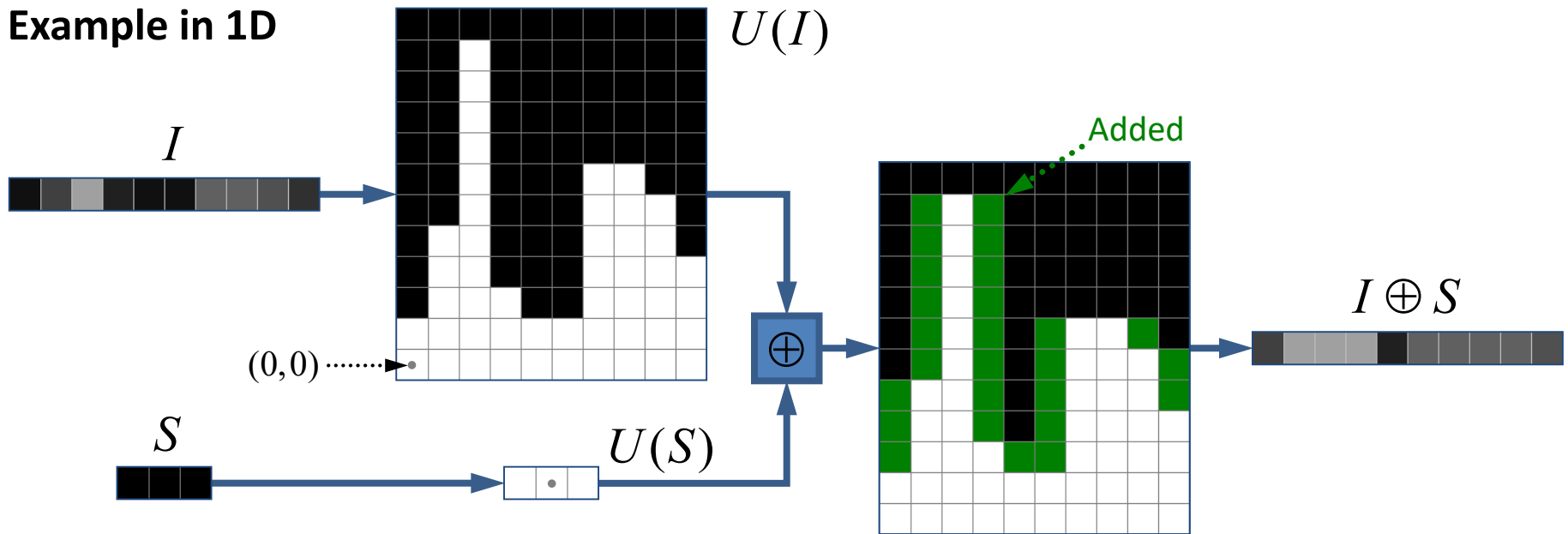
The landscape surface with the volume below is called the *umbra* of the image

# Dilation of gray-scale images

**Definition of gray-scale dilation:**  $I \oplus S = U^{-1} [U(I) \oplus U(S)]$

That is, the *binary dilation* of the umbra  $U(I)$  of *gray-scale image*  $I$  with the umbra  $U(S)$  of *gray-scale structuring element*  $S$ , turned back into gray-scale

**Example in 1D**

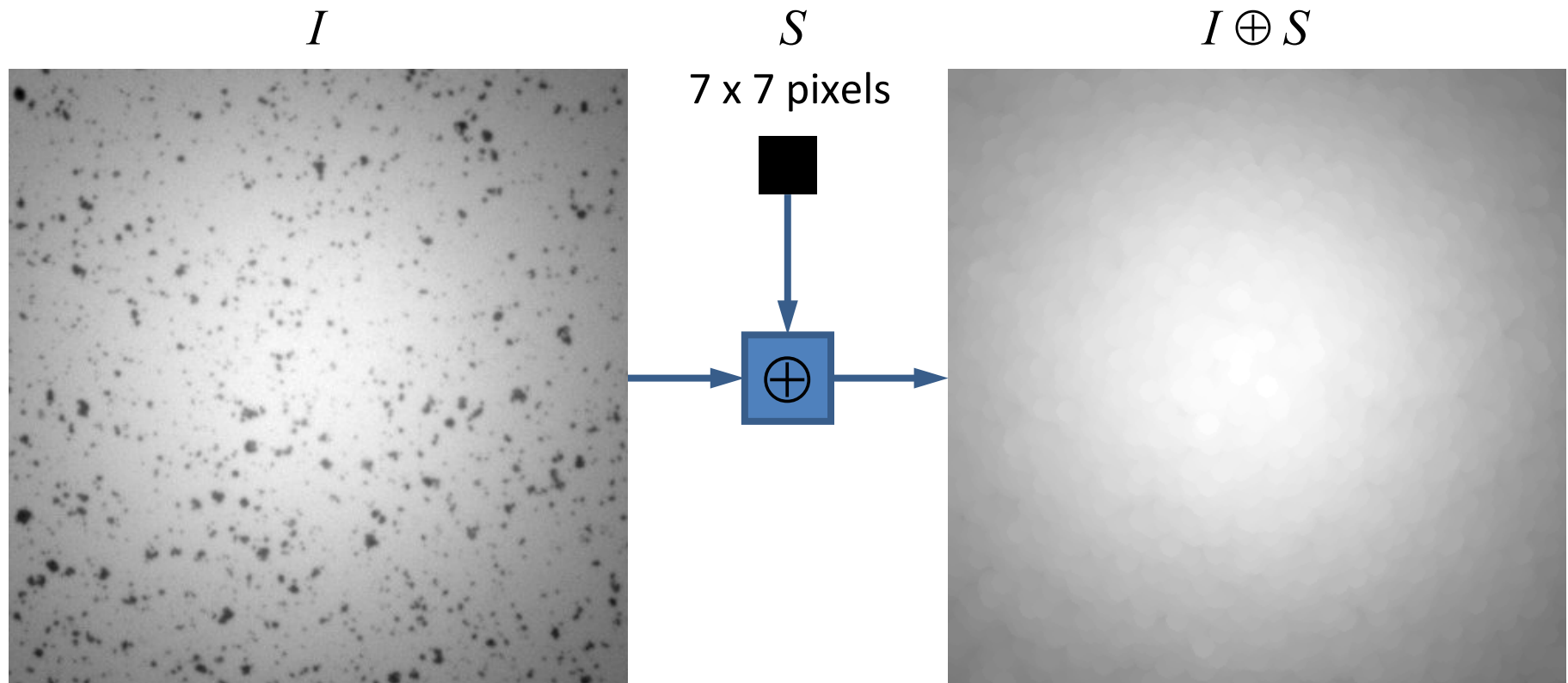


Any gray-scale  $S$  is possible but the flat one (as shown here) is used most often

# Dilation of gray-scale images

**Equivalent definition:**  $(I \oplus S)(\mathbf{x}) = \max_{\mathbf{p}} \{I(\mathbf{x} - \mathbf{p}) + S(\mathbf{p})\}$

For a flat and symmetrical structuring element this is simply local max-filtering

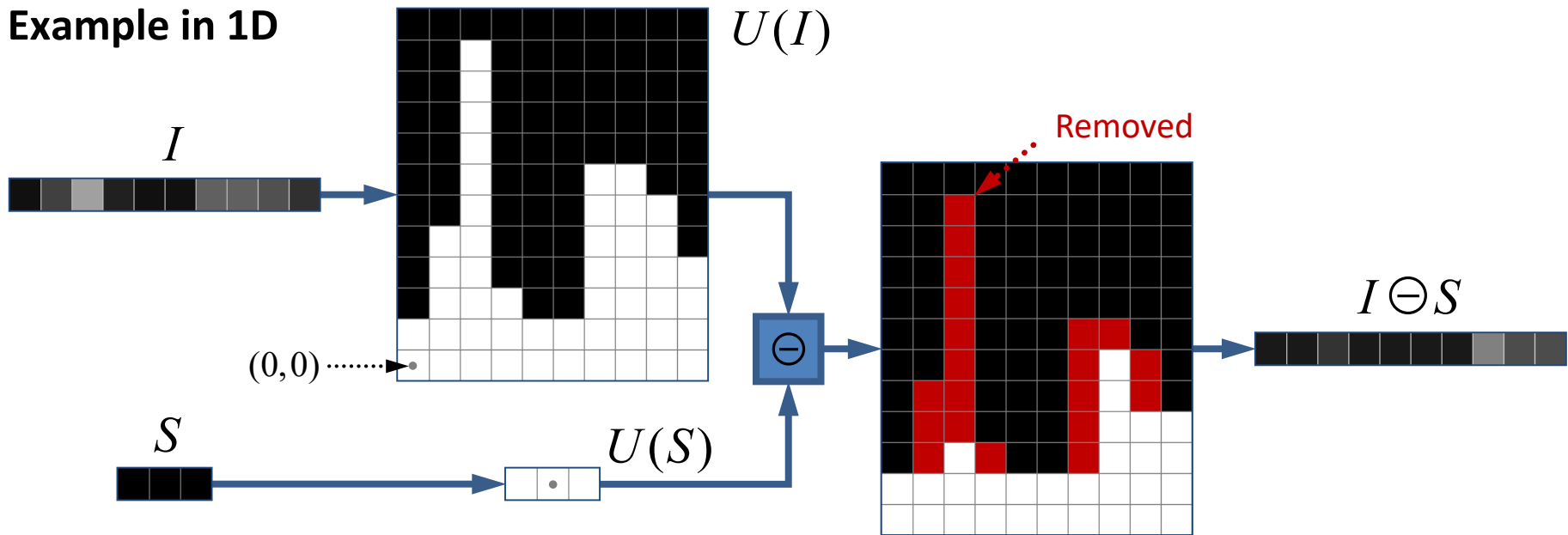


# Erosion of gray-scale images

**Definition of gray-scale erosion:**  $I \ominus S = U^{-1} [U(I) \ominus U(S)]$

That is, the *binary erosion* of the umbra  $U(I)$  of *gray-scale image*  $I$  with the umbra  $U(S)$  of *gray-scale structuring element*  $S$ , turned back into gray-scale

**Example in 1D**

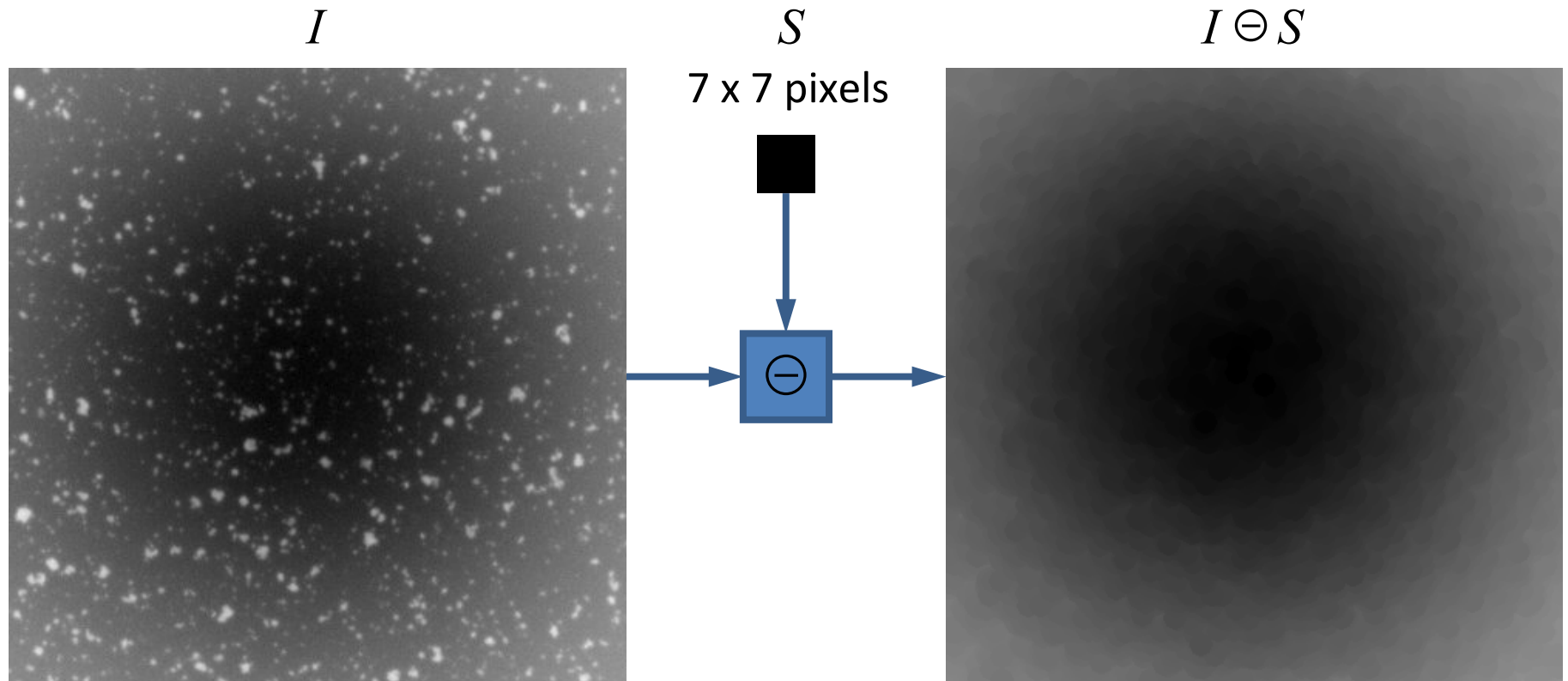


Any gray-scale  $S$  is possible but the flat one (as shown here) is used most often

# Erosion of gray-scale images

**Equivalent definition:**  $(I \ominus S)(\mathbf{x}) = \min_{\mathbf{p}} \{I(\mathbf{x} + \mathbf{p}) - S(\mathbf{p})\}$

For a flat and symmetrical structuring element this is simply local min-filtering

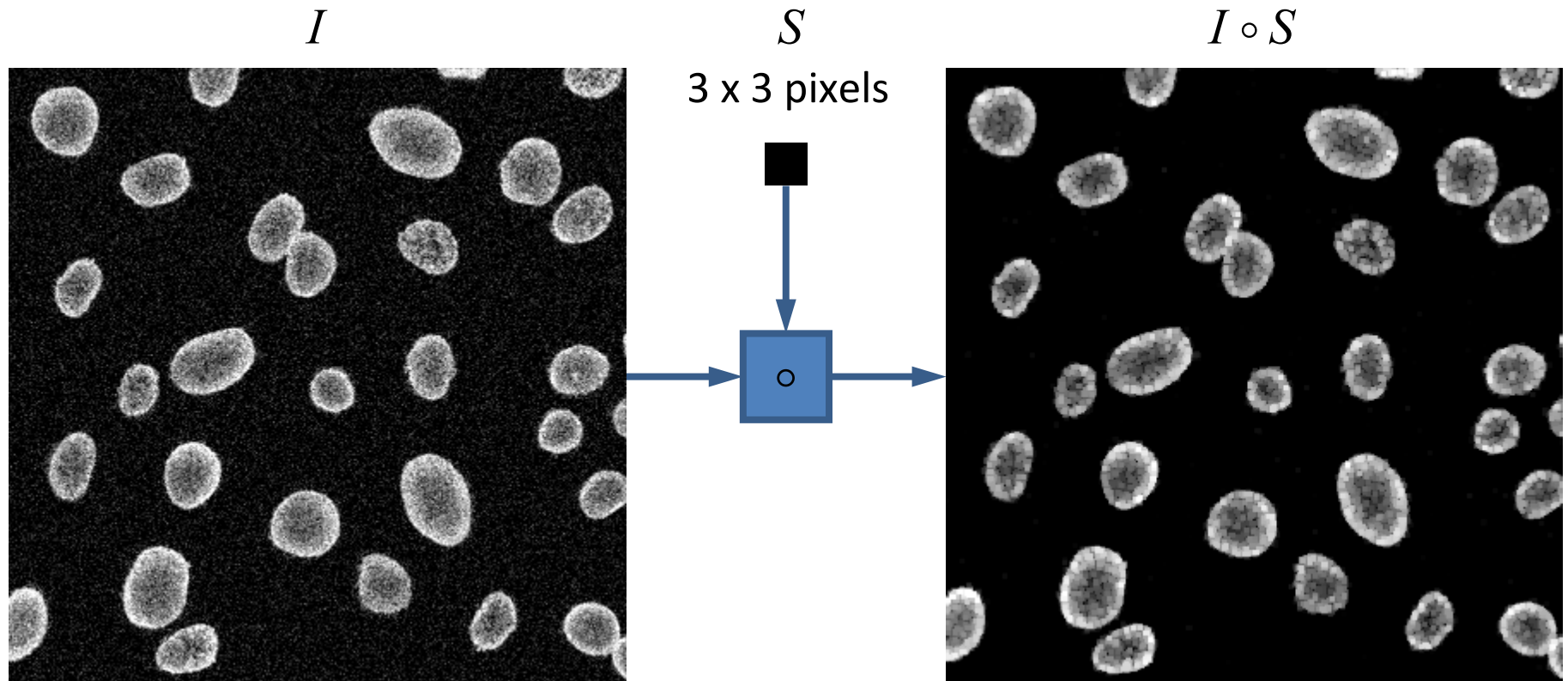




# Opening of gray-scale images

**Definition of gray-scale opening:**  $I \circ S = (I \ominus S) \oplus S$

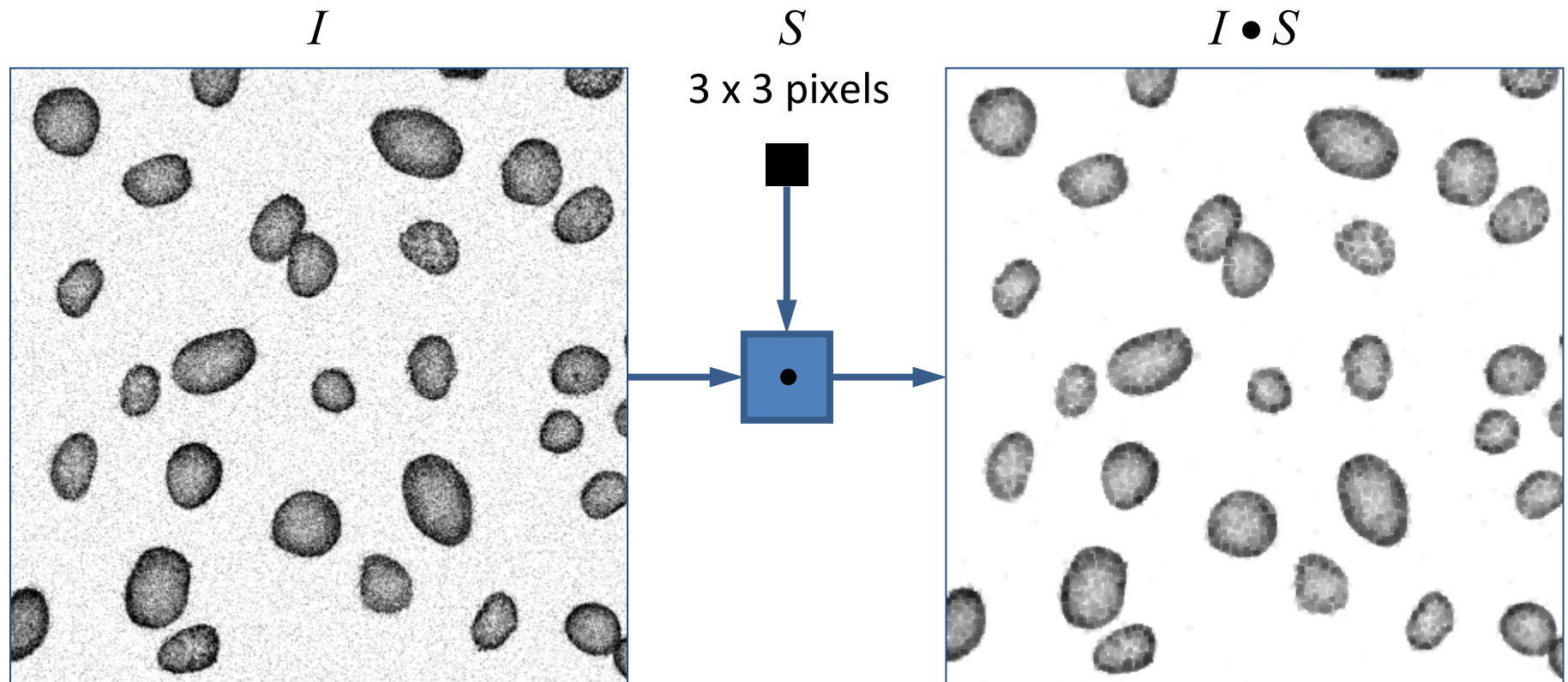
Gray-scale erosion and then gray-scale dilation with same structuring element



# Closing of gray-scale images

**Definition of gray-scale closing:**  $I \bullet S = (I \oplus S) \ominus S$

Gray-scale dilation and then gray-scale erosion with same structuring element



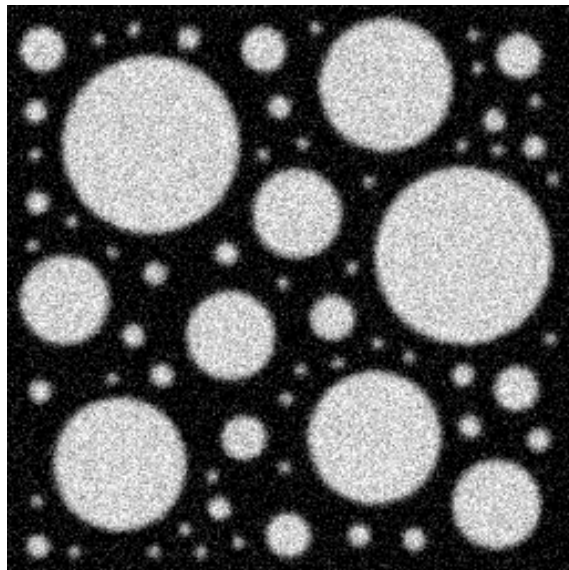
# Morphological smoothing of gray-scale images

---

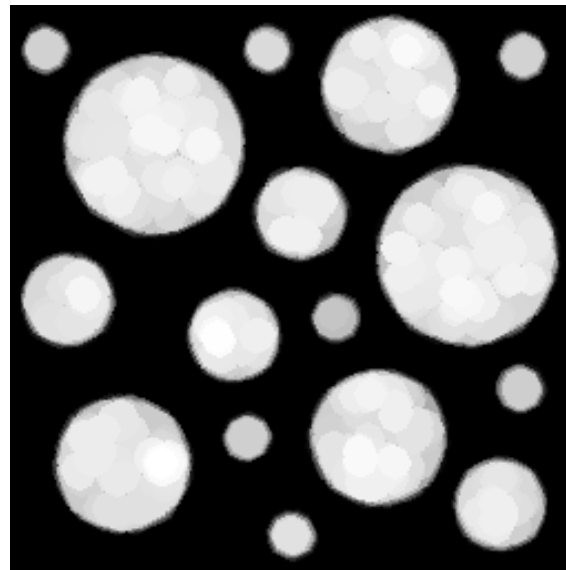
## Suppressing image structures of specific size (and shape)

- High-valued (bright) image structures are removed by gray-scale opening
- Low-valued (dark) image structures are removed by gray-scale closing

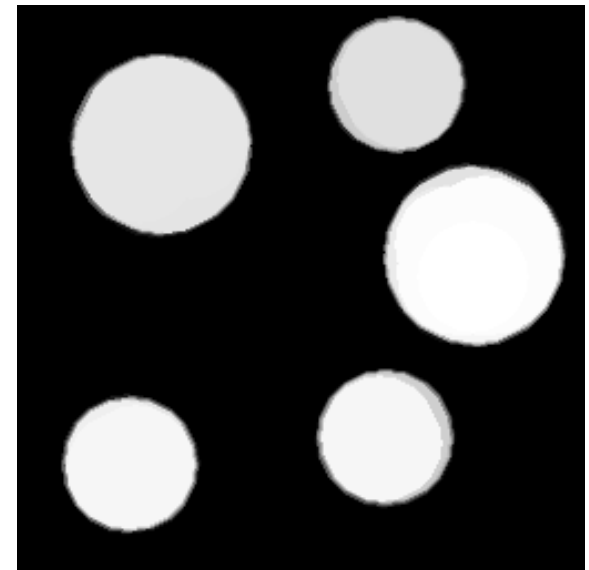
$I$



$I \circ S$  (radius = 7 pixels)



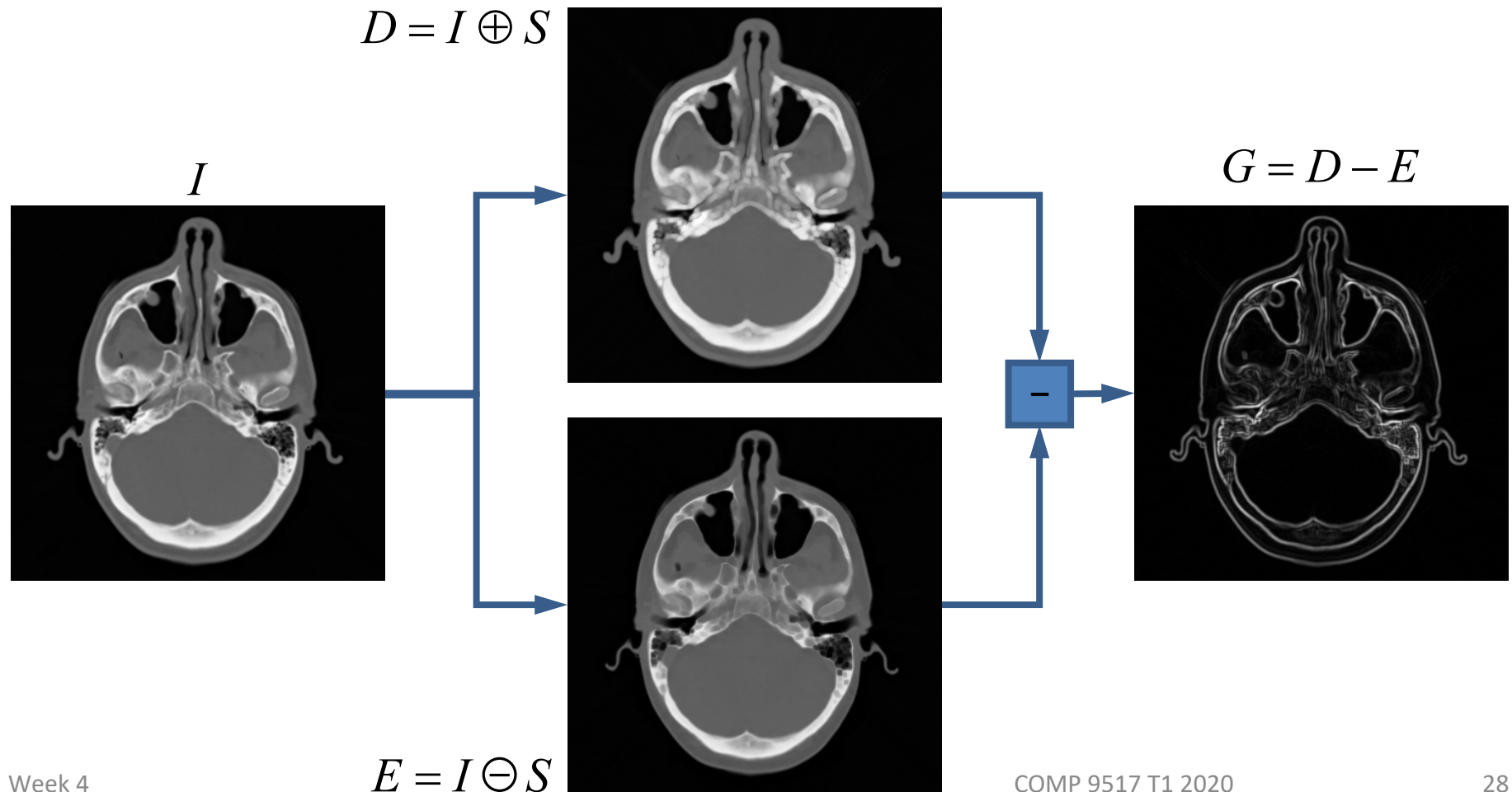
$I \circ S$  (radius = 25 pixels)



Behold the power of nonlinear filtering (not possible with linear filtering)

# Morphological gradient of gray-scale images

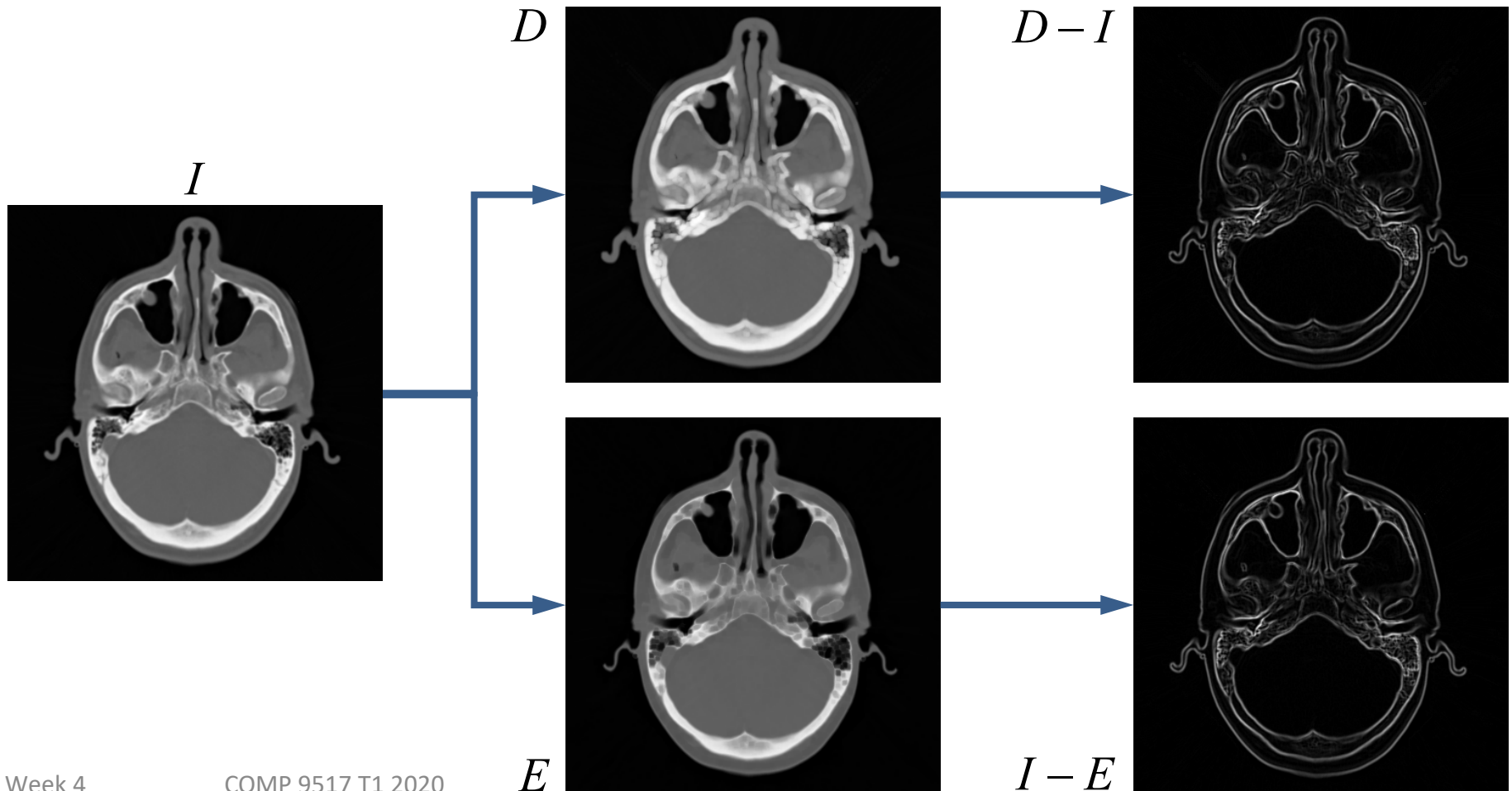
Difference between the dilated and the eroded image



# Morphological gradient of gray-scale images

29

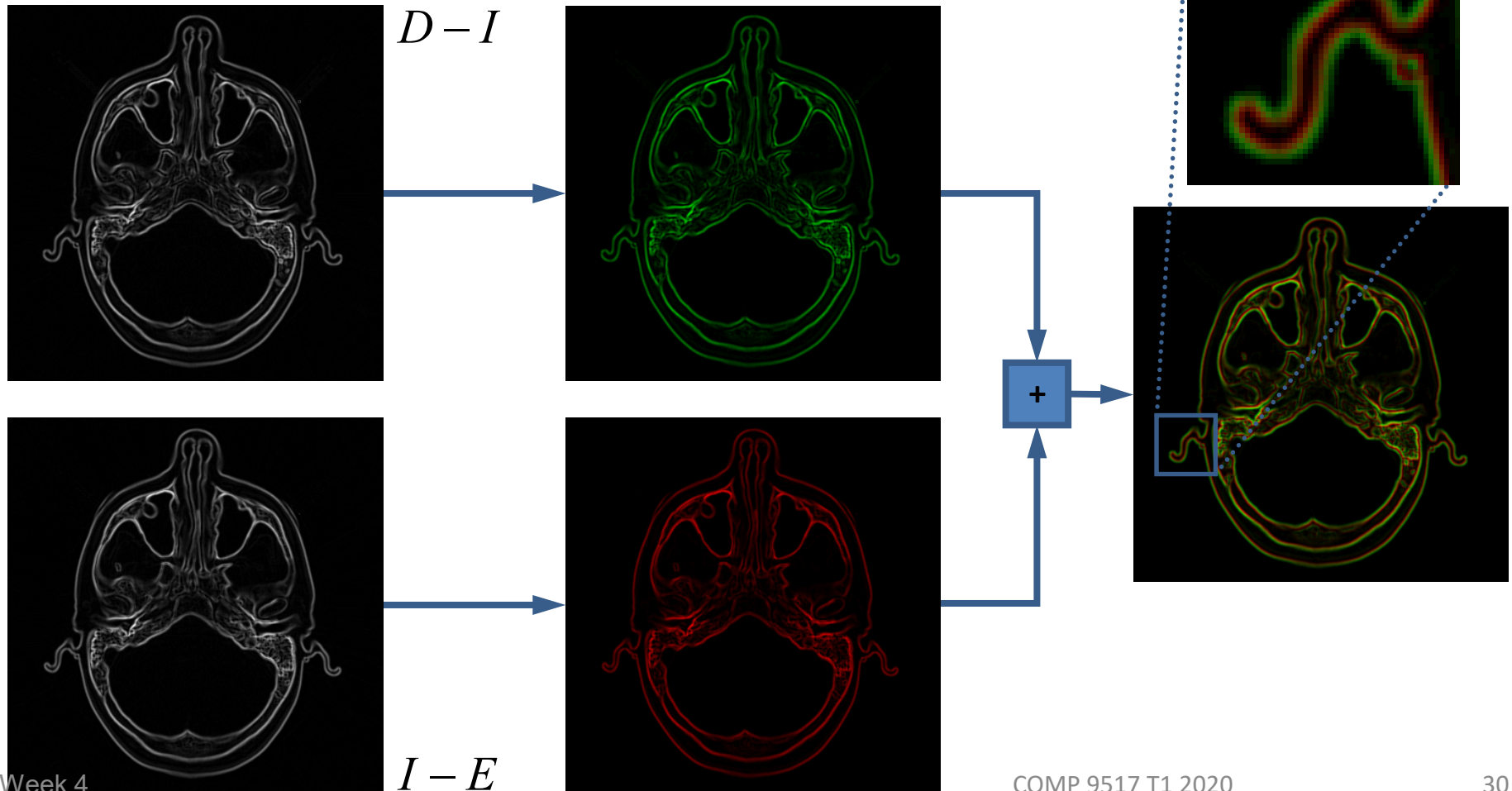
## Outer gradient and inner gradient





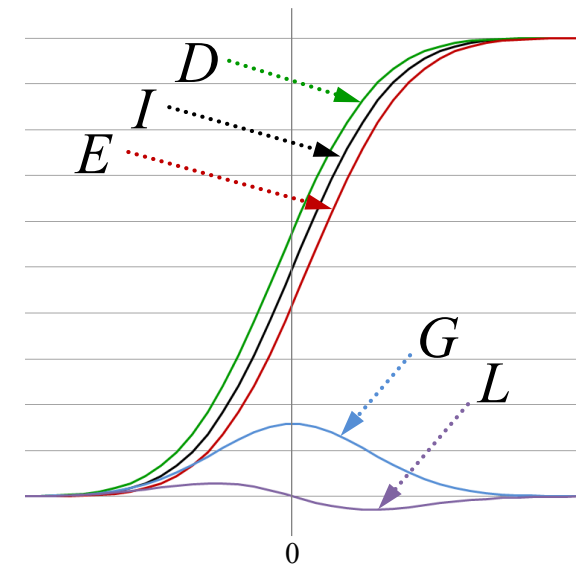
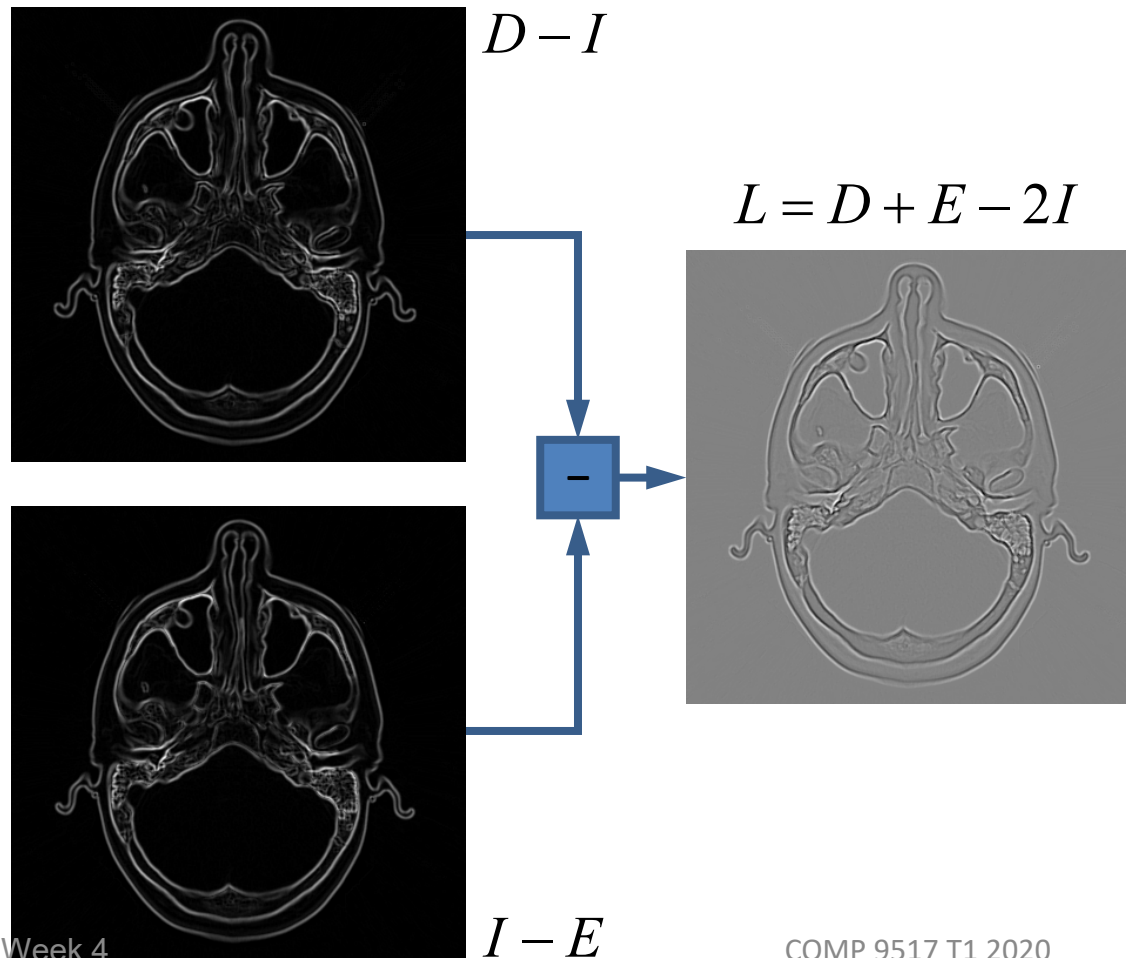
# Morphological gradient of gray-scale images

## Sum of outer gradient and inner gradient

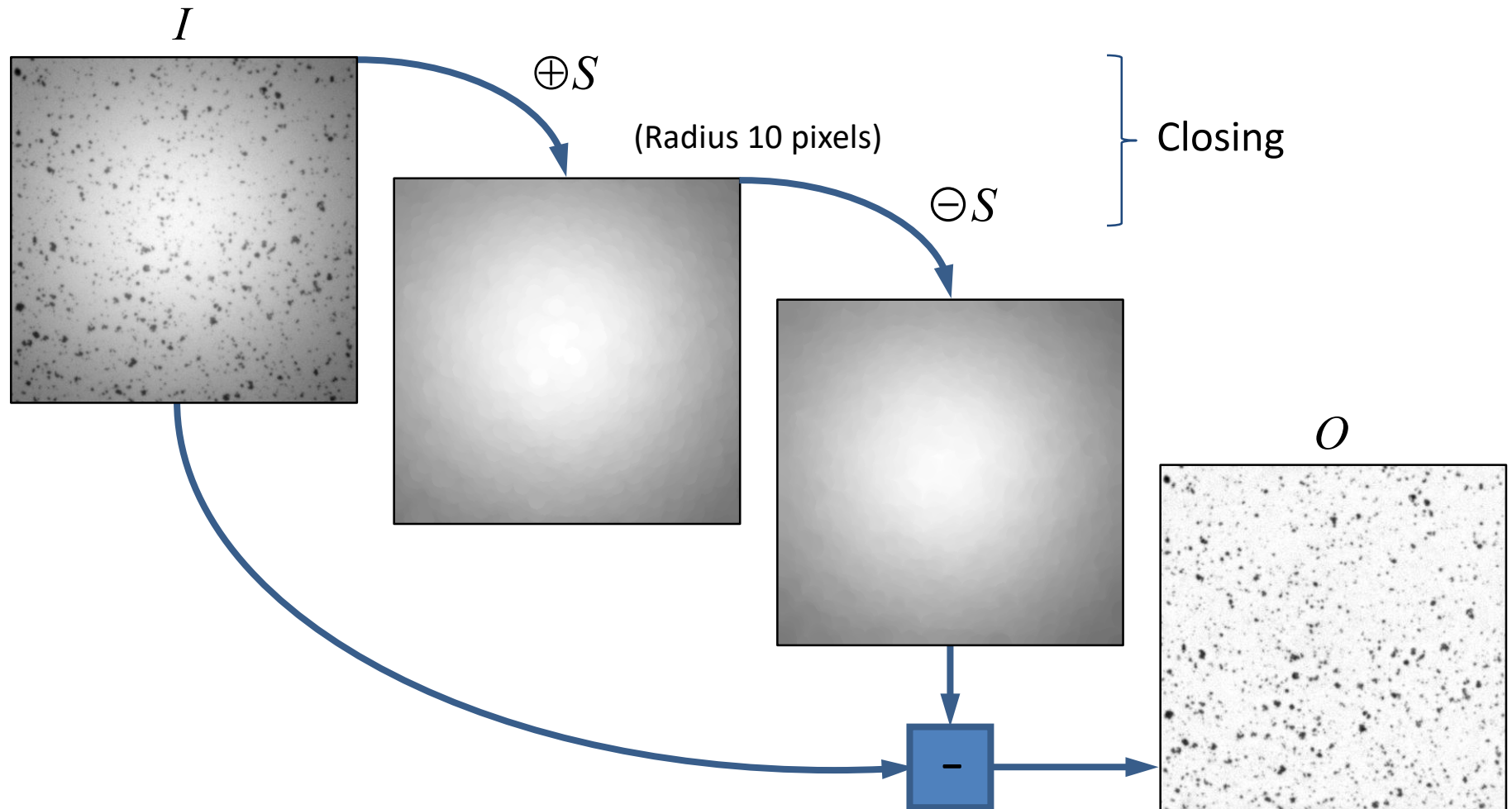


# Morphological Laplacean of gray-scale images

## Difference between outer gradient and inner gradient



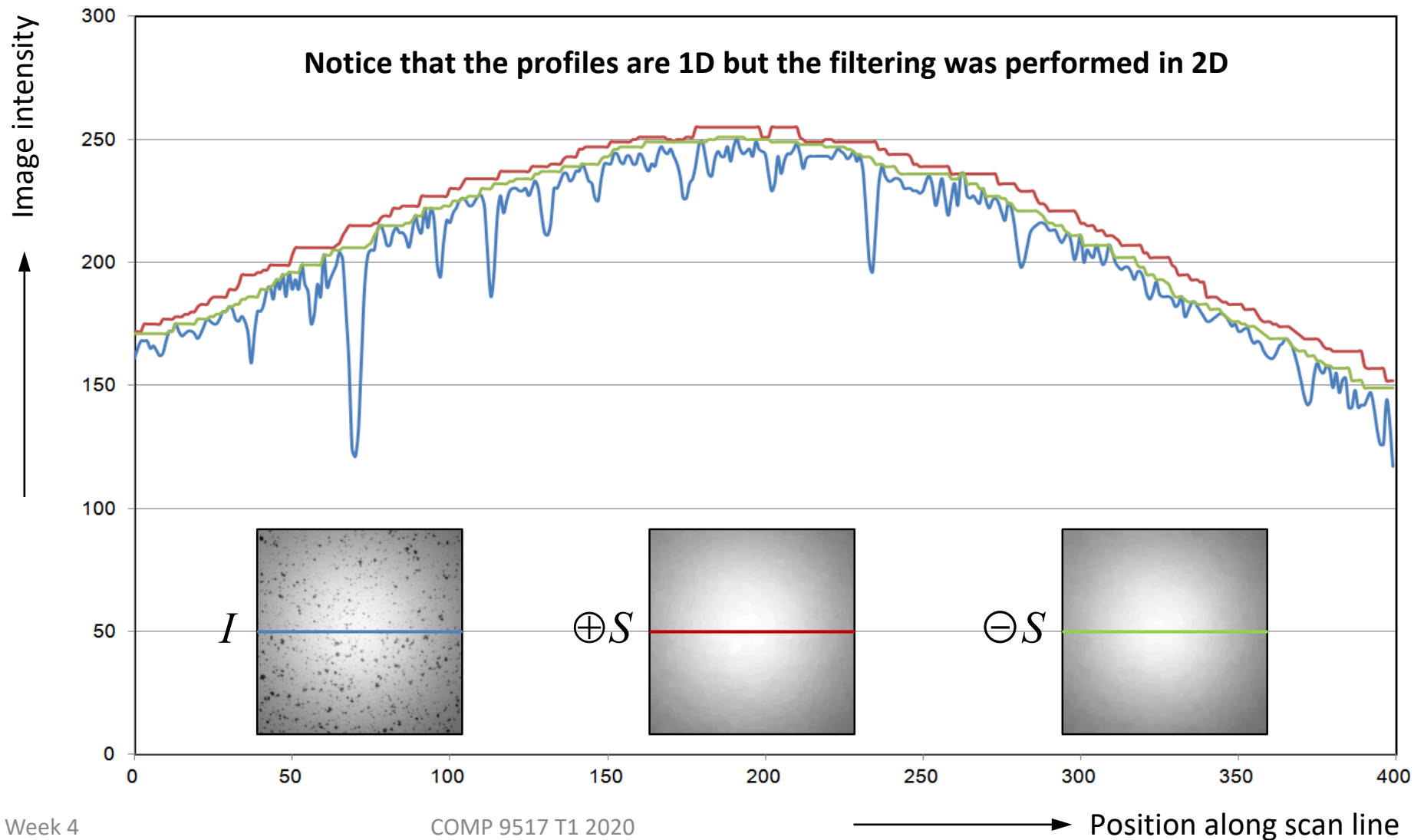
# Top-hat filtering of gray-scale images





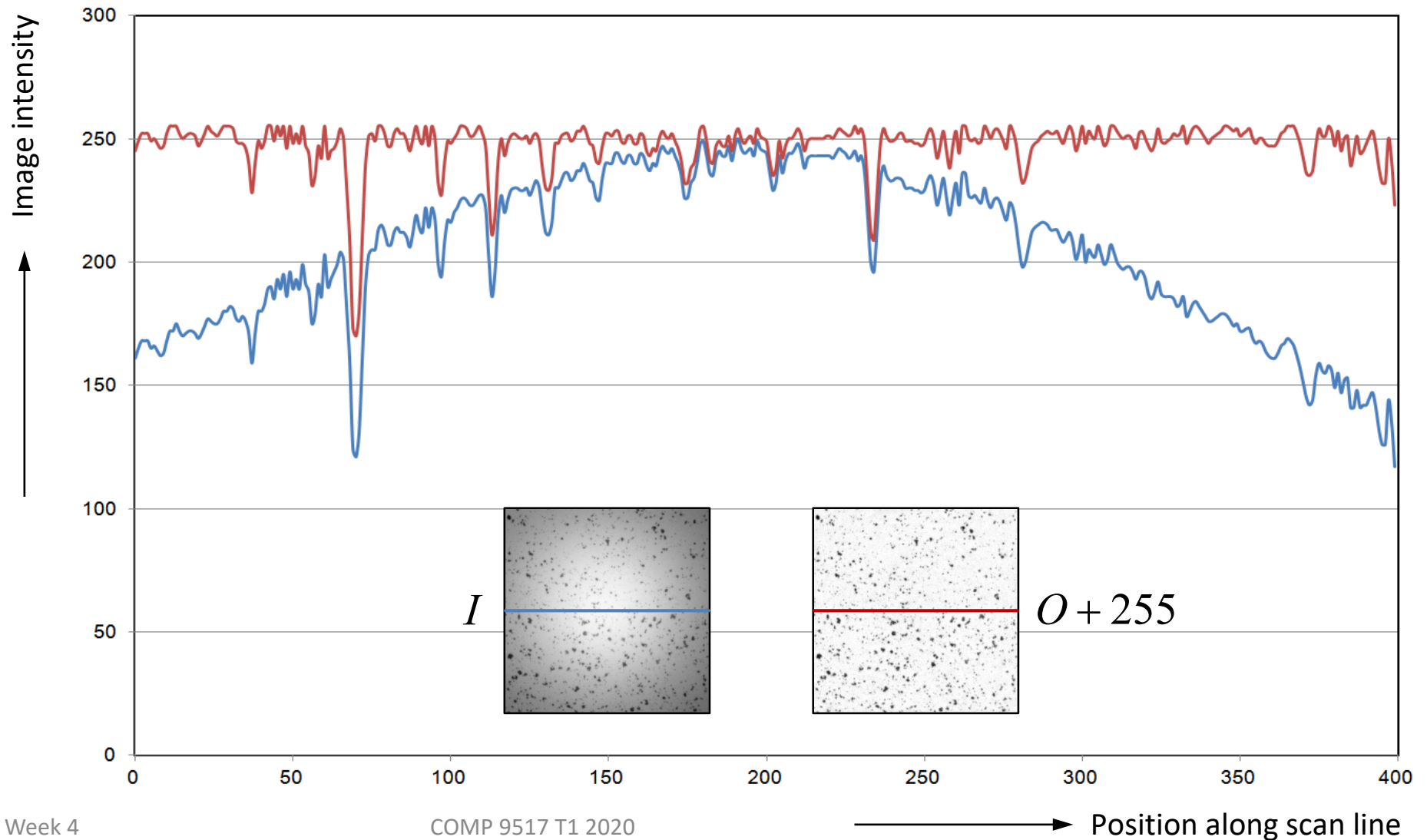
# Top-hat filtering of gray-scale images

33



# Top-hat filtering of gray-scale images

34



# Summary of mathematical morphology

---

## Powerful toolbox of methods around image segmentation

- **Gray-scale morphology for pre-processing**

- Removal of gray-scale noise

- Removal of background shading

- Removal of unwanted image structures

} Before segmentation

- **Binary morphology for post-processing**

- Closing holes in objects and background

- Finding the inner or outer outline of objects

- Detecting and separating touching objects

- Extracting representative shapes of objects

- Computing distances within and between objects

} After segmentation